1. m个人的成绩存放在数组score中，编写函数fun，它的功能是：
2. 求得低于平均分的人数
3. 求得低于平均分的分数
4. 将上述两种值带回主函数

int fun(int score[],int m, int below[]) //below[]用于存放低于平均分的分数

{

int i,k=0,aver=0; //k值记录人数

for(i=0;i<m;i++)

aver +=score[i];

aver /=m

for(i=0;i<m;i++)

{

if(score[i]<aver)

{

below[k]=score[i];

K++;

}

}

return k;

}

1. 请编写函数void fun(int x, int pp[], int \*n)，他的功能是：
2. 求出能整除x但不是偶数的各整数
3. 按从小到大的顺序存放在pp所指的数组中
4. 这些除数的个数通过形参n返回

void fun(int x,int pp[], int\*n)

{

int i =1,j=0,k=0,\*t=pp;

for(i=0;i<=x;i++)

{

if(i%2!=0)

{

t[j]=i;

j++

}

}

for(i=0;i<j;i++)

{

if(x%t[i]==0)

{

pp[k]=t[i];

k++;

}

}

\*n=k;

}

1. 写一个函数，输入一个四位数字，要求输出这四个数字字符，但每两个数字间空一个空格，如输入2021，输出’2 0 2 1’

#include<stdio.h>

#include<string.h>

int main()

{

int i,l;

char a[10];

gets(a);

l=strlen(a);

for(i=0;i<l;i++)

{

printf(“%c”,a[i]);

printf(“ ”);

}

return 0;

}

1. 要求使用指针，将一个3\*3的整形矩阵转置

#include<stdio.h>

void fun(int (\*p1)[3],int ( \*p2)[3]);

int main()

{

int p1[3][3]={0},p[3][3]={0};

int i=0,j=0;

for(i=0;i<3;i++)

{

for(j =0;j<3;j++)

{

scanf(“%d”,&p2[i][j]);

}

}

fun(p1,p2);

for(i=0;i<3;i++)

{

for(j=0;j<3;j++)

{

printf(“%d”,\*(\*(p1+i)+j));

}

printf(“\n”)

}

return 0;

}

void fun(int (\*p1)[3], int (\*p2)[3])

{

int i=0,j=0;

for(i=0;i<3;i++)

{

for(j=0;j<3;j++)

{

\*(p1[j]+i)= \*(p2[i]+j);

}

}

}

1. 定义一个结构体变量，包括年月日，计算某日在某年中是第几天，注意闰年问题

#include<stdio.h>

typedef strcut Date

{

int year;

int month;

int day;

} Date1;

int main()

{

int m,n=0,i,j,k,flag=0;

scanf(“%d %d %d ”,&Date1.year,&Date1.month,Date1.day);

if(Date1.year%4==0) //判断闰年问题

{

if(Date1.year%100!=0)

{

flag =1;

}

}

if(Date1.year%400==0)

{

flag=1;

}

for(i=1;i<Date1.month;i++) //天数的累加

{

switch(i)

{

case 1:

case 3:

case 5:

case 7:

case 8:

case 10:

case 12: m=31;break;

case 4:

case 6:

case 9:

case 11: m=30;break;

}

if(i==2)

{

if(flag)

{

m=29;

}

else m=28;

n+=m;

}

}

n+=Date1.day;

printf(“%d”,n);

return 0;

}

1. 有一磁盘文件“employee”，内存放职工的数据，每个职工数据包括职工姓名，职工号，性别，年龄，住址，工资，健康状况，文化程度。今要求将职工名，工资的信息单独抽出来另建一个简明的职工工资文件。

#include<stdio.h>

#include<stdlib.h>

#include<string.h>

typedef struct employee

{

char num[6];

char name[10];

char sex[2];

int age;

char addr[20];

int salary;

char health[8];

char class[10];

}em;

typedef struct emp

{

char name[10];

int salary;

}em\_case[10];

int main()

{

FILE \*fp1,\*fp2;

int i,j;

if((fp1=open(“employee”,”r”))==NULL)

{

printf(“cn not open file.\n”)

return 0; //exit(0);退出或者说是终止程序

}

printf(“\n NO. Name sex age addr salary health class\n”);

for(i=0;fread(&em[i],sizeof(struct employee),1,fp1)!=0;i++)

{

printf(“\n%4s%8s%4s%6d%10s%6d%10s%8s”,em[i].num,em[i].name,em[i].sex,em[i].age,em[i].addr,em[i].salary,em[i].health,em[i].class);

strcpy(em\_case[i].name,em[i].name);

em\_case[i].salary=em[i].salary;

}

printf(“\n\n \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*”);

if((fp2=fopen(“emp\_salary”,”wb”))=NULL)

{

printf(“cn not open file.\n”)

return 0;

}

for(j=0;j<i;j++)

{

if(fwrite(&em\_case[j],sizeof(struct emp),1,fp2)!=1)

printf(“error!”);

printf(“\n %12s%10d”,em\_case[j].name,em\_case[j].salary);

}

printf(“\n \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*”);

fclose(fp1);

fclose(fp2);

}